

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Možnosti kolaborativního programování v prostředích Matlab, Scilab a GNU Octave

Collaborative Programming in Matlab, Scilab and GNU Octave

Zadání bakalářské práce

Student: **Tomáš Žitník**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2601R013 Telekomunikační technika

Téma: Možnosti kolaborativního programování v prostředích MATLAB, Scilab
a GNU Octave
Collaborative Programming in MATLAB, Scilab a GNU Octave

Jazyk vypracování: čeština

Zásady pro vypracování:

Cílem práce je popsat možnosti zdarma poskytovaných nástrojů pro kolaborativní editaci zdrojového kódu s možností kompilace kódu na pracovních stanicích jednotlivých přispěvatelů v prostředích MATLAB, Scilab a GNU Octave.

1. Popište možnosti využití "Scilab Cloud API", "Octave Online" apod. pro možnosti výuky.
2. Proveďte rešerši zdarma poskytovaných nástrojů pro real-time kolaborativní editaci zdrojového kódu v multiplatformním prostředí Linux/MS Windows.
3. Popište možnosti propojení daného nástroje s verzovacím prostředím Git.
4. Popište možnosti využití daného nástroje pro koleborativní editaci kódu a jeho lokální kompilaci v prostředích MATLAB, Scilab a GNU Octave.

Seznam doporučené odborné literatury:


- [1] Šimeček, Jiří. *Webový nástroj pro kolaborativní editaci textů*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2018.
- [2] Ali Asghar Zafer. *NetEdit: A collaborative editor*. Master's thesis, Virginia Polytechnic Institute and State University, 2001. URL: <http://hdl.handle.net/10919/32127>.
- [3] Konečný, Martin. *Webový editor textů s podporou souběžné práce více uživatelů*. Diplomová práce. Brno: Vysoké učení technické, Fakulta informačních technologií, 2010. URL: <https://www.vutbr.cz/studenti/zav-prace/detail/34825>

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jan Skapa, Ph.D.**

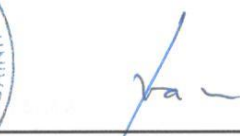
Datum zadání: 01.09.2019

Datum odevzdání: 30.04.2020



prof. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry





prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 15. května 2020

Žižka.....

Rád bych na tomto místě poděkoval svému vedoucímu bakalářské práce Ing. Janu Skapovi, Ph.D., za ochotný přístup, rady a připomínky.

Abstrakt

Cílem této bakalářské práce je popsat možnosti zdarma poskytovaných nástrojů pro kolaborační editaci zdrojového kódu s možností kompilace kódu na pracovních stanicích jednotlivých přispěvatelů v prostředí MATLAB, Scilab a GNU Octave. Za cíl je také kladeno pokusit se vhodně vybrat nástroj, který by mohl být užit přímo ve výuce. Takto vybraný nástroj by měl studentům poskytnout přímou interakci s vyučujícím a umožnit zapojení se přímo do řešeného problému v elegantní, pohodlné a srozumitelné podobě. Pro ozkoušení jednotlivých nástrojů bylo připraveno testování na dvou virtuálních stanicích v prostředí Virtualbox s operačním systémem Linux Mint a hostitelským systémem Windows 10.

Klíčová slova: Kolaborace, MATLAB, Scilab, GNU Octave, zdarma

Abstract

The aim of this thesis is to describe the possibilities of free tools for collaborative editing of source code with the possibility of compiling code on workstations of individual contributors in MATLAB, Scilab and GNU Octave. The aim is also to try to appropriately select a tool that could be used directly in the classroom. Such a tool should provide students with a direct interaction with teacher and enable them to participate directly in the problem solution in an elegant, comfortable and understandable form. For the testing of individual tools were prepared two virtual stations in the Virtualbox environment with Linux Mint and Windows 10 host system.

Keywords: Collaboration, MATLAB, Scilab, GNU Octave, free

Obsah

Seznam použitých zkratk a symbolů	8
Seznam obrázků	9
Seznam tabulek	10
Úvod	11
1 Úvod do verzovacího systému Git a práce se službou GitHub	12
1.1 Git	12
1.2 GitHub	13
1.3 Vytvoření GitHub účtu, instalace a konfigurace systému Git	14
1.4 Instalace Git pro Windows	14
2 Rešerše zdarma poskytovaných nástrojů (Desktop aplikace)	19
2.1 Atom	19
2.2 Gobby	26
2.3 Etherpad	31
3 Rešerše zdarma poskytovaných nástrojů (Web aplikace)	34
3.1 Overleaf	34
3.2 CryptPad	37
3.3 Meetingwords	40
3.4 FirePad	42
3.5 Conclave	44
4 Popis a možnosti využití dalších nástrojů pro výuku	46
4.1 Octave online	46
4.2 Scilab Cloud API	48
4.3 Matlab drive	49
5 Porovnání a výběr softwaru pro výuku ze zdarma nabízených řešení	51
Závěr	54
Literatura	55
Seznam příloh	59

Seznam použitých zkratek a symbolů

MATLAB	– MATrix LABoratory
repo	– repozitář
SCM	– software configuration management
CRDT	– Conflict-Free Replicated Data Type
URL	– Uniform Resource Locator
HTML	– Hypertext Markup Language
JSON	– JavaScript Object Notation
COBOL	– COmmon Business Oriented Language
CSS	– Cascading Style Sheets
PHP	– Hypertext Preprocessor
Sass	– Syntactically awesome style sheets
SQL	– Structured Query Language
TOML	– Tom’s Obvious, Minimal Language
XML	– Extensible Markup Language
YAML	– YAML Ain’t Markup Language
MML	– man-machine language
MIT	– Massachusetts Institute of Technology
GPL	– General Public License
SW	– Software
OS	– Operační systém

Seznam obrázků

1	Terminál aplikace GitBash pro systém Windows	14
2	Vytvoření nového repozitáře s možností prohlížení již vytvořených	15
3	Informace k vytvoření repozitáře	15
4	Náhled a editace repozitáře	16
5	Ukázka ovládacích prvků pro sdílení a změnu programovacího jazyka	20
6	Ukázka spouštění skriptu z Atomu s kompilací v Octave	23
7	Ukázka spouštění skriptu z Atomu s kompilací v MATLAB	23
8	Ukázka práce v Atomu s Latexem	24
9	Ukázka editoru Gobby	30
10	Ukázka editoru Etherpad	33
11	Ukázka editoru Overleaf	36
12	Ukázka editoru CryptPad	39
13	Ukázka editoru MeetingWords	41
14	Ukázka editoru Firepad	43
15	Ukázka editoru Conclave	45

Seznam tabulek

1	Příkazy balíčku Script a jejich klávesové zkratky [35]	22
2	Přehled Overleaf funkcí [15]	35
3	Porovnání vlastností editorů	51

Úvod

Spolupráce je poměrně velkým tématem dnešní doby a v programování, editaci a ladění zdrojových kódů, při spolupráci více lidí je využíváno nejrůznějších nástrojů, které toto umožňují. Tato práce se především zabývá průzkumem zdarma poskytovaných editorů, které by bylo možno využít ve výuce s ohledem na širokou funkcionalitu a uživatelské pohodlí. Práce se také snaží osvětlit a předvést základy verzovacích systémů a práce s nimi.

V první kapitole se snažím čtenáři přiblížit základní informace, co je Git, GitHub jejich instalaci, konfiguraci a zacházení s nimi.

V druhé kapitole se věnuji samotné rešerši, konkrétně tedy části klasických desktopových aplikací, které lze stáhnout, nainstalovat a provozovat přímo na osobním počítači.

V třetí kapitole se podíváme na druhou část rešerše, tedy webové aplikace. Projdeme si i Latexový editor Overleaf, který lze v základu používat zdarma, případně i placenou verzi s vyšší funkcionalitou.

Ve čtvrté kapitole uvádím přehled velice známých nástrojů, které jsou přímo podporovány výpočtovými programy typu MATLAB, Octave a Scilab.

V páté kapitole zužují výběr ze všech zkoumaných editorů, porovnávám funkcionalitu a hledám nejvhodnější editor pro výuku.

1 Úvod do verzovacího systému Git a práce se službou GitHub

1.1 Git

Git je bezplatný a open source distribuovaný systém pro správu verzí navržený tak, aby zvládal vše od malých po velmi velké projekty s rychlostí a účinností.

- **Větvení a slučování**

Jedná se o funkci Git, díky níž se odlišuje od téměř všech ostatních systémů se správou softwarové konfigurace (SCM), protože má svůj model větvení. Git umožňuje, aby uživatelé měli více lokálních větví, které mohou být na sobě zcela nezávislé. Vytváření, slučování a mazání těchto vývojových větví trvá několik sekund.

- **Kompaktnost a rychlost**

Git je rychlý. Téměř všechny operace jsou prováděny lokálně, což mu poskytuje obrovskou rychlostní výhodu v porovnání s ostatními centralizovanými systémy, které neustále musejí někde komunikovat se serverem. Byl postaven tak, aby pracoval na linuxovém jádře, což znamená, že od prvního dne musel efektivně pracovat s velkými úložišti. Git je psán v jazyce C, což snižuje režii běhů spojených s jazyky vyšší úrovně. Rychlost a výkon byly primárním designovým cílem už od začátku.

- **Distribuvanost** Jednou z vlastností distribuovaného SCM, včetně Git, je distribuce. To znamená, že místo zálohy aktuálního typu zdrojového kódu provedete „klon“ celého úložiště.

- **Datová bezpečnost**

Datový model, který Git používá, zajišťuje kryptografickou integritu všech bitů projektu. Každý soubor a změna jsou opatřeny kontrolním součtem a načteny pomocí svého kontrolního součtu při zpětném odhlášení. To zaručuje, že z verzovacího systému dostaneme přesně vždy to, co do něho nahrajeme.

- **Licence**

Git je vydán pod GNU General Public License verze 2.0

[1]

1.2 GitHub

Už víme, že Git je systém pro správu verzí, ale jedná se o nástroj příkazového řádku. Centrum, ve kterém jsou a převážně se dějí všechny věci, které se týkají Git revoluce, je tzv. Hub, nebo chcete-li GitHub. Zde vývojáři ukládají své projekty a najdete tu také síť s podobně smýšlejícími lidmi.

- **Repozitář**

Jinak také úložiště (obvykle zkráceno na „repo“) je místo, kde jsou uloženy všechny soubory pro konkrétní projekt. Každý projekt má své vlastní repo a můžete k němu přistupovat s jedinečnou adresou URL.

- **Forking (Větvení)**

Znamená vytvoření nového projektu založeného na jiném projektu, který již existuje. To je úžasná funkce, která výrazně podporuje další rozvoj programů a projektů. Pokud na GitHubu najdete projekt, k němuž byste chtěli přispět, můžete rozvětvit repo, provést požadované změny a revidovaný projekt vydat jako nové repo. Pokud se původní úložiště, které jste vytvořili pro vytvoření nového projektu, aktualizuje, můžete tyto aktualizace snadno přidat do své stávající větve.

- **Žádosti**

Po výběru repozitáře a provedení velké revize projektu může dotyčný uživatel zažádat, aby jej původní vývojáři poznali - možná dokonce zahrnuli jeho revizi do oficiálního projektu/repozitáře. Lze tak učinit vytvořením žádosti. Autoři původního repozitáře mohou vidět vaši práci a poté si vybrat, zda ji přijmou do oficiálního projektu. Kdykoli zadáte požadavek na vyžádání, GitHub vám poskytne perfektní médium pro komunikaci s vámi a údržbou hlavního projektu.

- **Sociální síť**

Aspekt sociální sítě GitHub je pravděpodobně jeho nejsilnější funkcí, která umožňuje projektům růst více, než jen o jakékoli jiné nabízené funkce. Každý uživatel v GitHubu má svůj vlastní profil, který funguje jako životopis a zobrazuje minulé práce konkrétního uživatele a příspěvky do jiných projektů prostřednictvím vyžádaných žádostí.

- **Seznamy změn**

Když na projektu spolupracuje více lidí, je těžké sledovat revize - kdo změnil co, kdy a kde jsou tyto soubory uloženy. GitHub se o tento problém stará tím, že sleduje všechny změny, které byly přesunuty do úložiště.

[2]

1.3 Vytvoření GitHub účtu, instalace a konfigurace systému Git

Uvedené postupy jsou pro operační systémy Linux Mint (18.3) a Windows 10

1.3.1 Vytvoření GitHub účtu

GitHub účet zdarma může vytvořit prakticky každý, kdo se zaregistruje standardním způsobem (jméno, heslo, email, ...) a nepotřebuje využívat dalších služeb, jako např. privátní repozitář (toto je jedna z funkcí, které jsou zpoplatněny).

Registraci lze provést zde: <https://github.com/>.

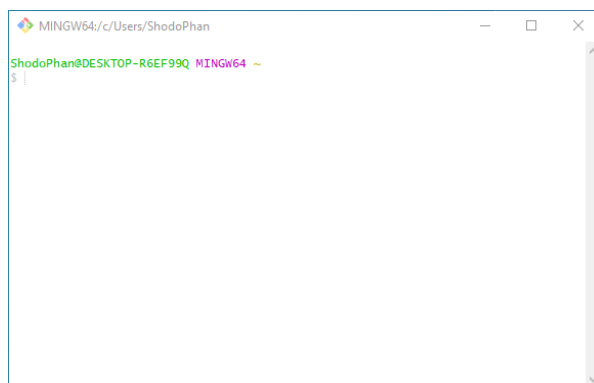
1.3.2 Instalace Git pro Linux

Pro instalaci Git pro operační systém Linux Mint lze využít správce balíčků, který je součástí samotné distribuce. Kombinací kláves **Ctrl+Alt+t** se otevře pracovní okno terminálů, kde lze zadávat příkazy pro práci se systémem a i výše zmiňovaným správcem balíčků. Do terminálu zadáme následující příkaz pro instalaci:

```
sudo apt install git
```

1.4 Instalace Git pro Windows

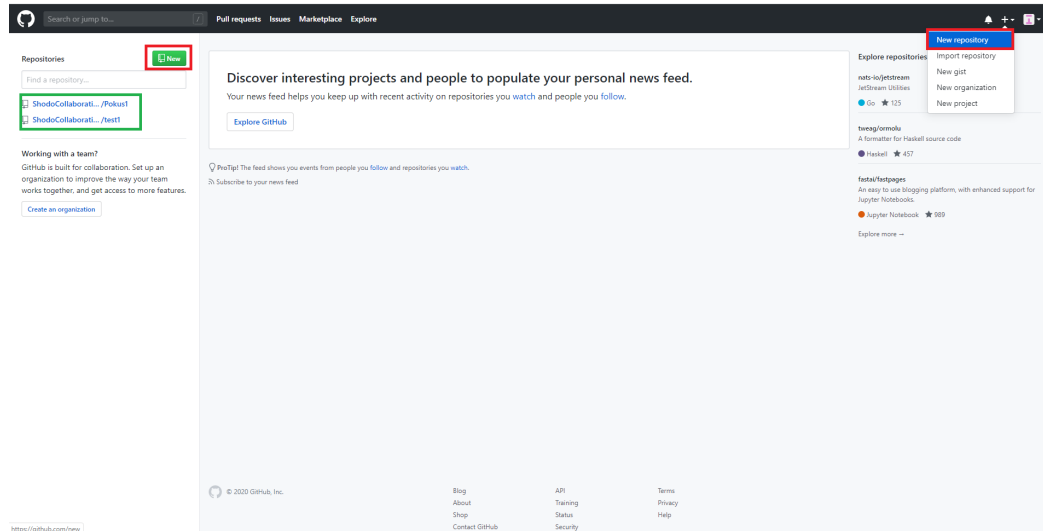
Pro instalaci Git pro operační systém Windows lze využít instalační soubor Git for Windows, který lze stáhnout na oficiálních stránkách <https://git-scm.com/download/win>, kde si lze vybrat z 32 nebo 64-bitové verze. Instalační soubor je vlastně průvodce, který umožňuje provést různá nastavení ještě před samotnou instalací (pro testovací účely bylo použito základního nastavení). Po spuštění nainstalované aplikace se otevře okno terminálu, kde lze zadávat příkazy jako v systému Linux.



Obrázek 1: Terminál aplikace GitBash pro systém Windows

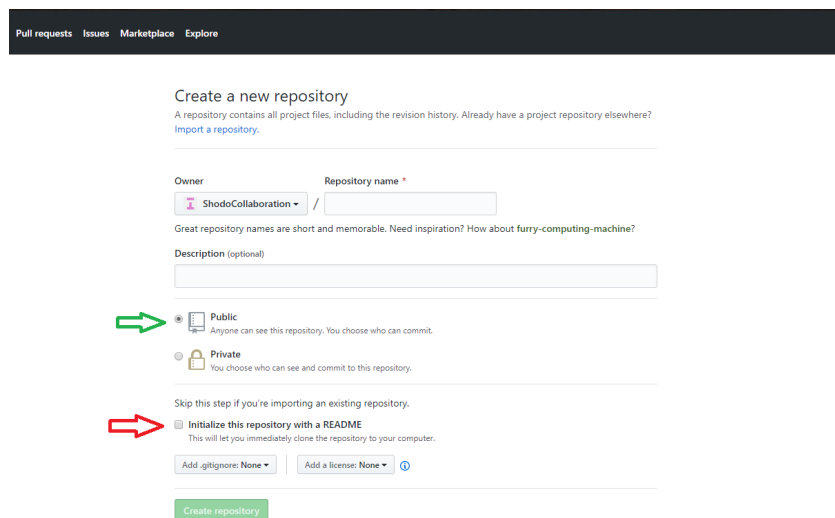
1.4.1 Vytvoření GitHub repozitáře

Jak lze vidět z obrázků č. 2, tak pro propojení GitHub služby a systému Git je potřeba nejprve vytvořit nový repozitář, kam budou nahrávány a ukládány soubory (**červený rámeček**). Lze také prohlížet a ručně editovat již vytvořené repozitáře (**zelený rámeček**).



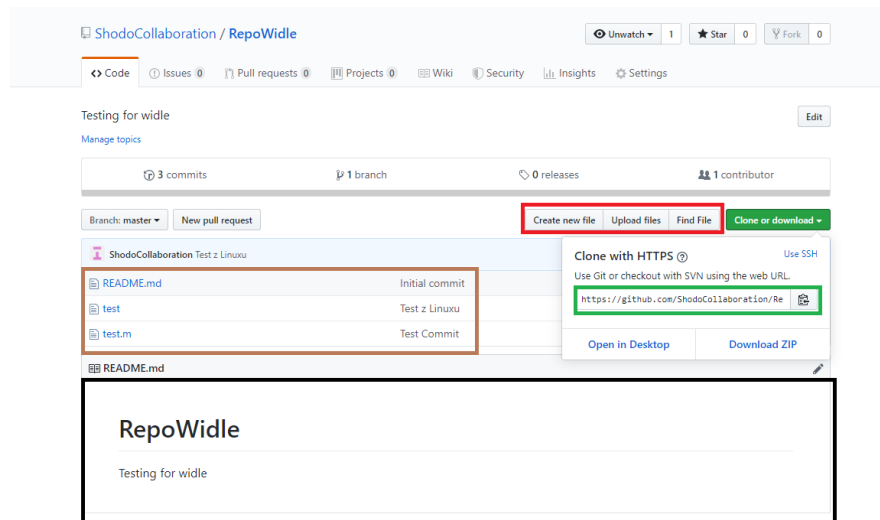
Obrázek 2: Vytvoření nového repozitáře s možností prohlížení již vytvořených

Pro vytvoření repozitáře je požadován název, popis, publicita repozitáře atd. Pro jednoduchost doporučuji zaškrtnout kolonku: „Initialize this repository with a README“ (**červená šipka**) na obrázku č. 3. Repozitáře jsou veřejně přístupné (**zelená šipka**) taktéž obrázek č. 3, privátní funkce je zpoplatněna. Do náhledu a editace repozitáře se lze dostat výběrem konkré-



Obrázek 3: Informace k vytvoření repozitáře

ního repozitáře na hlavní stránce, jako na obrázku č. 2. Po vstupu do repozitáře lze, zobrazeno na obrázku č. 4, provádět úpravy, např. ruční přidávání souborů (**červený rámeček**), prohlížení souborů (**hnědý rámeček**), čtení popisového souboru README (**černý rámeček**) a propojovat lokální složky přes odkaz s repozitářem (**zelený rámeček**).



Obrázek 4: Náhled a editace repozitáře

1.4.2 Nahrání souboru na GitHub pomocí terminálu v Linuxu

Pro nahrání souboru bude potřeba mít pro jednoduchost a přehlednost jednu konkrétní složku, kam budou ukládány projekty, určené k nahrání do GitHub repozitáře. Pro jednoduchost doporučuji domovský adresář `Home/"Vaše složka s~projektem"`. Kombinací kláves **Ctrl+Alt+t** se spustí terminál, do kterého je potřeba zadat následující příkazy:

```
cd "cesta ke složce se soubory, pro nahrání"
```

- pomocí příkazu `cd` se lze přesunout do složky, ve které se nachází projekt, který chcete nahrát do GitHub repozitáře
- pomocí příkazu `cd ..` se lze přesunout do nadřazeného adresáře

```
git init
```

- tento příkaz provede inicializaci lokální složky jako GitHub repozitáře

```
git add . nebo git add "Název souboru"
```

- tento příkaz přidá všechny soubory v aktuální složce <.> nebo jeden konkrétní soubor <Název souboru>

```
git commit -m "Váš první commit"
```

- tento příkaz potvrdí přidané soubory, včetně komentáře

```
git remote add origin "url Vašeho GitHub repozitáře"
```

- tento příkaz přidá URL adresu, která reprezentuje GitHub repozitář. Zde bude nahrán lokální repozitář (soubory v lokální složce)
- adresu GitHub repozitáře lze získat z webové služby GitHub (viz obrázek 1.5-zelený rámeček)

```
git remote -v
```

- ověření nového URL

```
git push origin master
```

- nahraje soubory z lokálního repozitáře (složky) do GitHub repozitáře, který byl specifikován
- poté se objeví výzva na zadání vašeho přihlašovacího jména a hesla pro GitHub
- po zadání údajů se provede odeslání souborů do GitHub repozitáře a od tohoto okamžiku jsou soubory přístupné k prohlížení a další práci pomocí webové služby GitHub

[3]

1.4.3 Stažení souborů z GitHub repozitáře

Klávesovou kombinací **Ctrl+Alt+t** lze otevřít terminál a zadat následující příkazy:

```
cd "cesta ke složce pro stažení repozitáře"
```

- pomocí příkazu `cd` se lze přesunout do složky, kam mají být soubory s GitHub repozitáře staženy
- pomocí příkazu `cd <..>` se lze přesunout do nadřazeného adresáře

```
git clone "url adresa GitHub repozitáře"
```

- adresu GitHub repozitáře lze získat z webové služby GitHub (viz obrázek 1.5-zelený rámeček)
- po provedení příkazu budou soubory z GitHub repozitáře nakopírovány a stáhnuty do lokálního adresáře

[34]

1.4.4 Smazání GitHub repozitáře

Na GitHub webové službě, po vstupu do hlavní stránky konkrétního repozitáře, lze postupovat následovně:

- pod názvem repozitáře kliknout na Settings
- lze nalézt „Danger Zone“, ve které lze vybrat možnost „Delete this repository“
- po zobrazení upozornění je nutné zadat jméno repozitáře
- potvrdit smazání repozitáře

2 Rešerše zdarma poskytovaných nástrojů (Desktop aplikace)

2.1 Atom

Atom je desktopový textový editor, jehož back-end je založený na Electronu¹ (oficiálně známém jako Atom Shell), s možností editace zdrojového kódu na platformách MS Windows, Linux a macOS. V základní „čisté“ verzi také obsahuje vestavěný ovládací prvek pro verzovací systém Git. Aplikace však umožňuje rozšíření funkcionality, případně i personalizace prostředí (nejčastěji vzhled), pomocí rozšiřujících balíčků (pluginů). Pro front-end byl použit CoffeeScript, JavaScript, Less a HTML.[30]

2.1.1 Požadavky

- Internet
- GitHub účet - nepovinné
(pouze pro možnost použití integrované funkcionality pro nahrávání do verzovacího systému)

2.1.2 Balíčky

Jako většina ostatních textových editorů, které umožňují rozšiřitelnost a konfigurovatelnost, Atom nabízí možnost instalovat balíčky a motivy třetích stran, k přizpůsobení funkcionality a vzhledu editoru. Balíčky lze nainstalovat, spravovat a publikovat pomocí vestavěné utility nazývané „Atom’s package manager“.[30, 31]

Většina rozšiřujících balíčků je pod licencí „Free software license“ a jsou vyvíjeny a udržovány komunitou.

2.1.3 Podporované programovací jazyky

Editor v základu dokáže aplikovat zvýraznění syntaxe pro následující programovací jazyky a soubory:

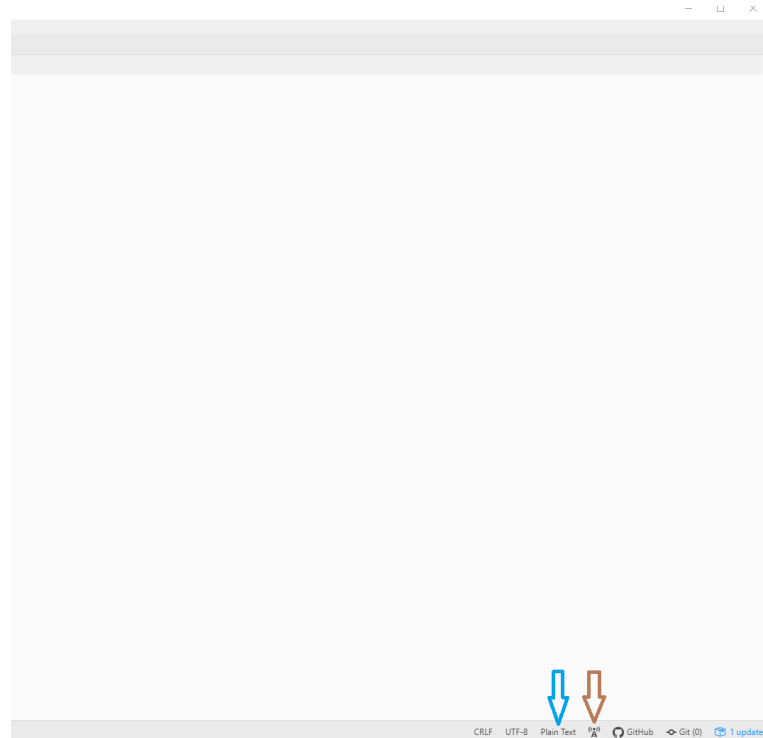
C, C++, C#, Clojure, COBOL, CSS, CoffeeScript, D, GitHub Flavored Markdown, Go, HTML, Java, JavaScript, JSON, Julia, Less, Make, Mustache, Objective-C, Perl, PHP, Property lists, Python, Racket, Ruby, Ruby on Rails, Sass, Scala, Shell, SQL, TOML, TypeScript, XML, YAML, MML a další.[4, 30]

Přepínání jazyků je možné provádět ovládacím prvkem ([modra šipka](#)), jak je ukázáno na obrázku č. 5.

¹Jedná se o framework, který umožňuje použití aplikací pro více platforem pomocí Chromia a Node.js.[5]

2.1.4 Balíček Teletype

Asi nejvíce zajímavý balíček z pohledu této práce je balíček Teletype, který přidává do Atomu funkcionalitu, která umožňuje vývojářům a obecně všem, kdo s editorem pracují, sdílet své pracovní prostředí s ostatními členy týmu a spolupracovat na editaci kódu v reálném čase. Ovládací prvek (hnědá šipka) je vidět na obrázku č. 5.



Obrázek 5: Ukázka ovládacích prvků pro sdílení a změnu programovacího jazyka

Spojení funguje následujícím způsobem:

- Nejprve se připojíte na server tvůrců Atomu, kde lze vidět, kdo je otevřen kolaboraci.
- Potom budete sdílet kód s ostatními spolupracovníky přímo, tedy peer-to-peer.
- Není zde žádný centralizovaný server, který by měl možnost odposlouchávat, co jste právě stiskli na klávesnici. Teletype navíc používá WebRTC k šifrování komunikace mezi všemi spolupracovníky.

[6]

- Instalace Teletype:

a) **Linux terminál:**

- * Nejprve stáhněte a naistalujte Atom ve verzi 1.22 nebo vyšší
- * Do terminálu zadejte příkaz:

```
apt install teletype
```

b) **GUI (i pro MS Windows):**

- * Nejprve stáhněte a naistalujte Atom ve verzi 1.22 nebo vyšší
- * Spusťte Atom
- * Otevřete „**Settings View**“ pomocí kombinace kláves **Ctrl + ,**
- * Napište teletype do vyhledávače
- * Klikněte na tlačítko „Install“

[31]

2.1.5 Možnost propojení daného nástroje s verzovacím systémem Git a webovou službou GitHub

Atom ve svém základu obsahuje balíček pro integraci Git a GitHub ihned po instalaci. Umožňuje standardní vytváření větví (branches), repozitářů, změnu fází vývoje a úprav, commit, pull, push a další.

Pro více informací o možnostech práce s Gitem a GitHubem přímo v textovém editoru, doporučuji si projít dokumentaci k balíčku na následujícím odkazu:

<https://flight-manual.atom.io/using-atom/sections/github-package/>

2.1.6 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Atom, jak už bylo zmíněno, nabízí nepřehledné množství rozšiřujících balíčků, s jejichž pomocí lze přizpůsobit funkcionalitu editoru potřebám konkrétního uživatele. Pro kompilaci zdrojových souborů na lokálním počítači je nutné mít nainstalovaný příslušný software, ve kterém si přejeme kompilaci provést. Důležité je také zmínit, že funkčnost byla testována na OS Linux Mint, protože balíček Script, který tuto funkcionalitu umožňuje, pravděpodobně používá na pozadí syntaxe, které umí zpracovat terminál v Linuxu, což ve Windows způsobuje chyby typu, že daný příkaz neexistuje apod.

Tabulka 1: Příkazy balíčku Script a jejich klávesové zkratky [35]

Příkaz	macOS	Linux/Windows
Run	cmd+i	shift+ctrl+b
Run by Line Number	shift+cmd+j	shift+ctrl+j
Run Options	shift+cmd+i	shift+ctrl+alt+o
Run with profile	shift+cmd+k	shift+ctrl+alt+b
Close View	esc/ctrl+w	esc
Kill Process	ctrl+c	ctrl+q

Znovu připomínám to, že určení SW, který bude daný skript kompilovat, lze učinit výběrem jazyku v pravém dolním rohu Atomu. Pokud daný jazyk chybí, je nutné stáhnout a doinstalovat balíček, jako další rozšíření Atomu.

Atom balíčky pro jednotlivá prostředí

- Matlab a GNU Octave

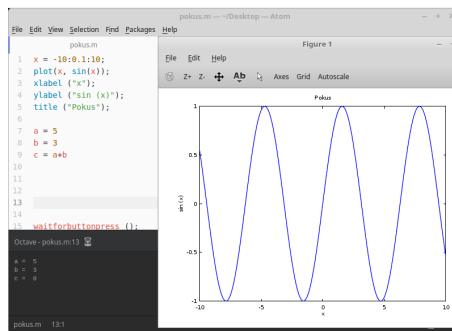
- script - balíček pro kompilaci a spouštění skriptů přímo z Atomu.
- language-matlab - balíček pro podporu jazyků pro MATLAB A Octave.

Pozn. Bohužel se ukázalo, že Atom má problém s voláním MATLABU na OS Linux. Tento problém lze řešit přidáním MATLABU do „**PATH**“ nebo případně zadat do terminálu příkaz `sudo apt install matlab-support`, který stáhne a nainstaluje balíček pro podporu komunikace MATLABU s Atomem.

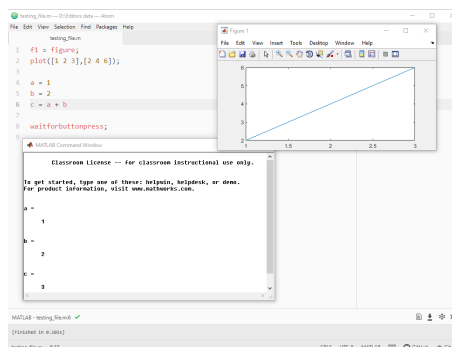
- Scilab

- scilab-language - balíček pro podporu jazyka Scilab

Pozn. Bohužel v současné době balíček script nepodporuje kompilaci pro prostředí Scilab a alternativu se mi nepodařilo najít.



Obrázek 6: Ukázka spouštění skriptu z Atomu s kompilací v Octave

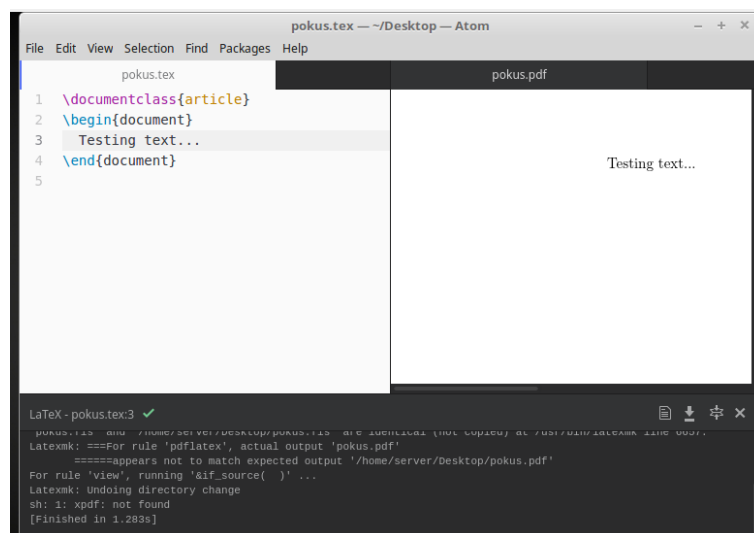


Obrázek 7: Ukázka spouštění skriptu z Atomu s kompilací v MATLAB

2.1.7 Funkce navíc

Jako speciální funkcionalitu navíc, která byla testována opět na OS Linux, bych zde rád zmínil psaní textů a provázání Atomu s Latexem. Pro funkcionalitu je samozřejmě nutné mít nainstalovanou distribuci Latexu (doporučuji TexLive) a následující balíčky pro Atom:

- script - balíček pro kompilaci a spouštění skriptů přímo z Atomu.
- language-latex - balíček pro zvýraznění syntaxe
- latextools - balíček rozšířené podpory pro Latex
- pdf-view - balíček, který umožní zobrazit výsledné pdf po každé kompilaci. (není nutný pro funkčnost)



Obrázek 8: Ukázka práce v Atomu s Latexem

Nutno uvést, že většina balíčků po nainstalování obsahuje i text dokumentace, případně odkazy na ní v souborech README, které lze nalézt následujícím postupem:

- V Atom editoru zmáčkneme klávesovou kombinaci **Ctrl+čárka** - tím se dostaneme do nastavení editoru
- V menu vybereme položku „**Packages**“
- Zde můžeme filtrovat balíčky podle jména, případně dvojklikem otevřít konfiguraci samotného balíčku
- Po posunutí posuvníků na samotný konec lze objevit sekci s názvem **README**, případně názorné ukázky použití daného balíčku atd.

- Vzhledem k tomu, že se jedná o Open Source, tak je možno si i v Atomu přímo zobrazit zdrojový kód tlačítkem „**View code**“, které lze nalézt pod názvem daného balíčku ihned po rozkliknutí.

2.2 Gobby

Gobby je vcelku zajímavým řešením pro uživatele, kteří z různých důvodů vyžadují vlastní provoz serveru pro možnosti sdílení a editaci textů v reálném čase mezi jednotlivými přispěvateli a nechtějí k tomu využívat služeb třetích stran. Tento SW umožňuje jak přímé propojení klient-klient, tak také možnost klient-server. Propojení s vlastním serverem, který je pod správou konkrétní organizace, nabízí jednu podstatnou výhodu oproti využívání služeb někoho jiného pro sdílení a editaci. Tou výhodou je jistota, že editované dokumenty a texty, které jsou sdílené mezi uživateli, nebudou posílány přes uzly poskytovatelů těchto služeb, nýbrž jenom přes soukromý server nebo přímo na klienta. Tím se zamezí případnému shromažďování a odposlechu dat. Další výhodou je, že provoz vlastního serveru lze přizpůsobit pro zabezpečení sítě organizace apod.

2.2.1 Požadavky

- Připojení k Internetu
- Balíček Gobby (pro klientské systémy)
- Balíček Infinoted (pro dedikovaný serverový systém) - nepovinný
pozn. Balíčky už obsahují základní prerekvizity, které jsou vypsány zde[7]

2.2.2 Funkčnost

- Běží na OS Windows, Linux a Mac OS X
- Plnohodnotné možnosti úpravy textu, včetně zvýraznění syntaxe
- Neomezená editace v reálném čase
- Integrovaný skupinový chat
- Group Undo (Undo neovlivňuje změny od vzdálených účastníků)
- Barevné rozlišení kurzorů a výběrů vzdálených účastníků
- Plná Unicode podpora
- Zeroconf² podpora (zatím pouze pro Linux)
- Šifrovaný přenos dat včetně perfektního dopředného utajení (PFS)
- Relace mohou být chráněny heslem
- Sofistikované řízení přístupu se seznamy řízení přístupu (ACLs)
- Vysoce konfigurovatelný dedikovaný server (včetně pluginů)

[7]

- Server a klienti umožňují bez složitého nastavování komunikovat jak pomocí IPv4, tak především v čím dál více preferované IPv6 na výchozím portu 6523

Pro více informací ohledně pluginů a dalších více specifických funkcí doporučuji přečíst oficiální dokumentaci:

<https://github.com/gobby/gobby/wiki/Dedicated-Server>

2.2.3 Propojení

Počítače ve stejné síti (LAN) jsou díky Gobby propagovány formou seznamu v Gobby klientovi na OS Linux. Pro OS Windows a Mac OS X není tato možnost zatím k dispozici. Klávesovou kombinací **Ctrl+t** nebo výběrem možnosti „**Connect to Server**“ a zadat (pokud známe) název počítače (hostname) nebo IP adresu. Připojení jak na sever, tak přímo na klienta lze chránit heslem, případně zabezpečení pomocí TLS.

²Soubor technologií pro automatické vytvoření použitelného síťového spojení, mezi propojenými zařízeními.

2.2.4 Instalace

- Windows

- Pro instalaci na OS Windows stačí stáhnout instalační soubor (Installer) z oficiální stránky (Soubor obsahuje jak Gobby klienta, tak i dedikovaný server infinoted)

<https://github.com/gobby/gobby/wiki/Download>

- Instalace je klasické spuštění souboru .exe a nainstalování do příslušného adresáře
- Pro spuštění Gobby klienta stačí spustit soubor **Gobby-0.5.exe**, který je umístěn do *<Vaše struktura adresářů/Gobby-0.5/bin>*. Po spuštění lze provést spoustu různých nastavení jako např. nastavení hesla pro připojení vzdálených účastníků, změna barvy a jména účastníka a další.
- Pro spuštění serveru je nutno využít příkazovou řádku (Cmd) v režimu správce. Zde je nutné se navigovat do stejného adresáře, jako v předchozím případě a zadat příkaz:

- * Pro nezabezpečené spojení

```
infinoted-0.6.exe --security-policy=no-tls
```

- * Pro zabezpečené spojení

```
infinoted-0.6.exe --create-key --create-certificate -k key.pem -c cert.pem
```

- tímto se vytvoří klíč a certifikát

- Následně lze server spustit zadáním příkazu

```
infinoted-0.6.exe -k key.pem -c cert.pem -password=heslo
```

- parametr pro heslo není nutné zadávat, pokud heslo nepožadujeme, ale vřele ho doporučuji používat

- Linux

- Pro instalaci na OS Linux máme dvě možnosti instalace:

- * Gobby klient - pro instalaci nám postačí otevřít terminál **Ctrl+ Alt + t** a zadat příkaz:

```
sudo apt-install gobby
```

Tím se nám nainstaluje Gobby klient, který spustíme zadáním následujícího příkazu do terminálu.

```
gobby
```

Klient je obdobný jako u Windows a lze v něm taktéž provést spoustu různých nastavení, jako např. nastavení hesla pro připojení vzdálených účastníků, změna barvy a jména účastníka a další.

- * Gobby server - pro instalaci Gobby serveru opět využijeme terminál a zadáme příkaz

```
sudo apt-get install infinoted
```

Poté je server možné spustit ve dvou režimech:

- Nezabezpečené připojení

```
infinoted -security-policy=no-tls
```

- Zabezpečené připojení

```
infinoted --create-key --create-certificate -k key.pem -c cert.  
pem
```

- tímto se vytvoří klíč a certifikát

- Následně lze server spustit zadáním příkazu

```
infinoted -k key.pem -c cert.pem -password=heslo
```

- parametr pro heslo není nutné zadávat, pokud heslo nepožadujeme, ale opět všude ho doporučuji používat

[8, 9, 10]

2.2.5 Licence

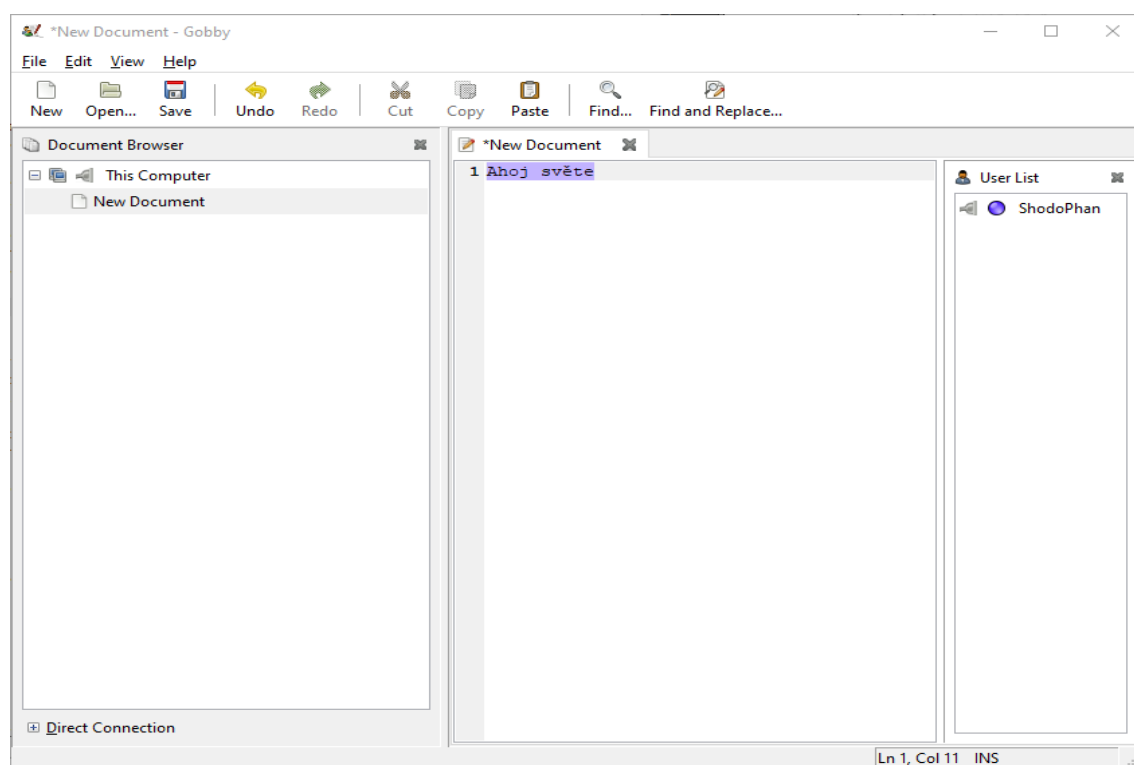
Tento SW je distribuován pod licencí GPLv2+ a ISC.

2.2.6 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Editor není v současné době možné přímo propojit s verzovacím systémem. Lze tak učinit ne přímo přes příkazovou řádku.

2.2.7 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Přímá kompilace není možná, ale vzhledem k možnostem editoru ukládat do zvolených adresářů, včetně možnosti pojmenování a uložení souborů v typech pro tato kompilační prostředí, tak není problém si to přes jeden pracovní adresář nastavit. Přínosem také je, že Gobby editor vždy pracuje s kopií daného dokumentu, která nemění originál uložený lokálně (platí především pro sdílení). Výhodou je to, že pokud by sdílený soubor kdokoli smazal, tak se nevymaže originál lokálně uložený, ale pouze pracovní kopie, která se sdílí.



Obrázek 9: Ukázka editoru Gobby

2.3 Etherpad

Etherpad je webový editor, který umožňuje spolupracovat na úpravách dokumentů v reálném čase, podobně jako ten, který běží v prohlížeči.

Všechny instance poskytují přístup ke všem datům prostřednictvím dobře zdokumentovaného rozhraní API a podporují import / export do mnoha hlavních formátů výměny dat. Dále existuje velké množství pluginů, které uživateli umožní přizpůsobit instanci tak, aby vyhovovala jeho potřebám.

K užívání Etherpadu se nemusí instalovat server a samotná aplikace Etherpad. Lze si vybrat jeden z veřejně dostupných, které zřídili ostatní lidé z celého světa, což může působit velice rizikově vzhledem k nedostatku informací ohledně serveru, přes který komunikace probíhá.

Po vyzkoušení jsem zde také objevil historii úprav všech uživatelů, rozlišování autorů pomocí barev a také je zde možnost chatu, který funguje v rámci projektu mezi všemi uživateli, kteří spolupracují na daném projektu.[11]

2.3.1 Požadavky

- node.js (10.13.0 nebo vyšší)[12]

2.3.2 Funkčnost

Editor běží na lokálním počítači, ale přístup k samotné aplikaci je možný provést pouze přes webový prohlížeč. V podstatě se dá říct, že editor funguje jako klasická webová stránka, která je hostovaná na nějakém hostitelském serveru (v našem případě lokálním počítači).

- Editoru umožňuje klasické základní úpravy textu (velikost písma, atd.)
- import³/export ve formátu: Etherpad, HTML, Plain text
- Zobrazení historie úprav
- Uložení aktuální úpravy
- Nastavení jazyka a fontu
- Sdílení dokumentu

2.3.3 Propojení

Propojení mezi dvěma a více spolupracovníky lze pomocí možnosti sdílení (umístěno v pravém horním rohu editoru). Vygeneruje se Link pro sdílení: `http://127.0.0.1:9001/<název Padu>`. Jak si lze všimnout, tak adresa 127.0.0.1 (localhost) je zde zobrazena proto, protože Etherpad

³Import lze rozšířit pomocí instalace AbiWord. Odkaz na návod k instalaci lze najít přímo v aplikaci Etherpadu v záložce import/export

běží z mého pohledu na mém lokálním počítači. Pro ostatní to však znamená, že budou muset znát IP adresu mého počítače a link upravit, například takto: `http://192.168.10.25:9001/<název Padu>`. Kdyby totiž zadali původní vygenerovanou adresu, tak se jim sice zobrazí nový editor (Pad) se stejným názvem, ale ke sdílení, jako takovému nedojde, protože se adresou localhost budou odkazovat na své lokální počítače.

2.3.4 Instalace

a) Linux terminál:

- Instalace prerekvizit:

```
sudo apt install git curl python libssl-dev pkg-config build-essential
```

- Instalace Node.js:

```
wget https://nodejs.org/dist/v6.9.2/node-v6.9.2-linux-x64.tar.xz
tar xJf node-v6.9.2-linux-x64.tar.xz
sudo mkdir /opt/nodejs/ && mv node-v6.9.2-linux-x64 /*/opt/nodejs>
echo "PATH=$PATH:/opt/nodejs/bin" >> ~/.profile
```

- Klonování binárních souborů pro Etherpad

```
sudo mkdir /opt/etherpad
sudo chown -R $(whoami).$(whoami) /opt/etherpad
cd /opt/etherpad
git clone git://github.com/ether/etherpad-lite.git
```

- Spuštění Etherpadu

```
cd opt/etherpad/etherpad-lite/bin/run.sh
```

[13]

b) Windows:

1. Stáhnout instalační balíček z oficiální stránky:

<https://etherpad.org/#download>

2. Extrahujte balíček do kterékoli adresáře vašeho počítače
3. Spustíte soubor **start.bat** a do webového prohlížeče zadejte adresu:

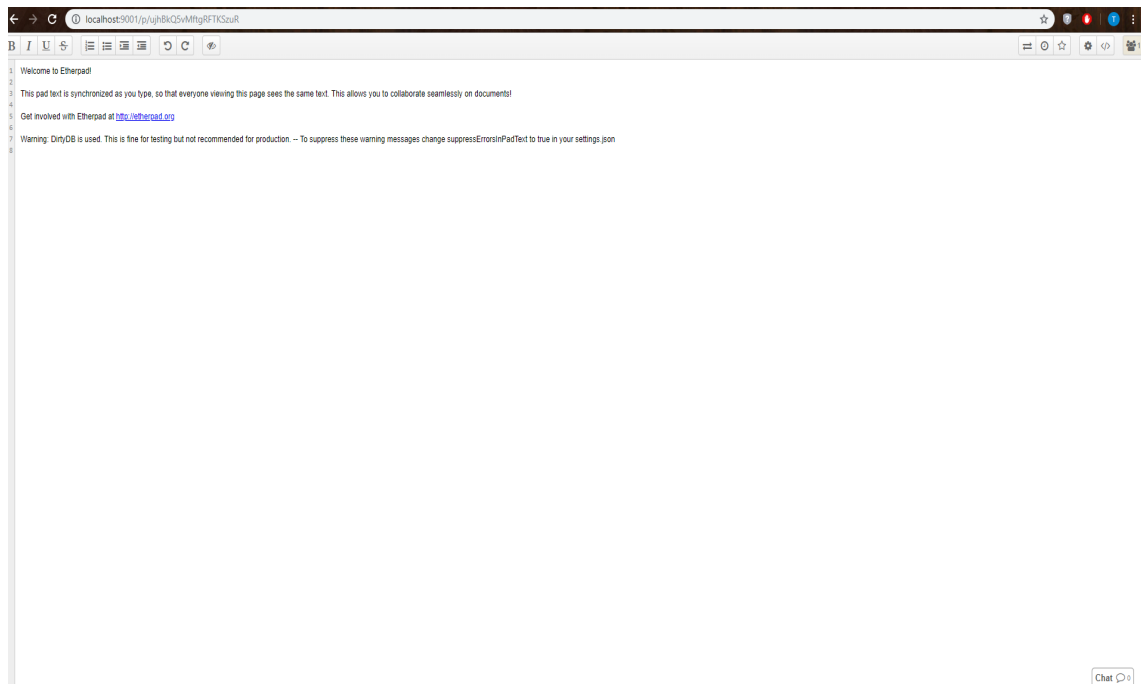
<http://localhost:9001>

2.3.5 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Editor neumožňuje propojení se systémy pro správu verzí prostřednictvím samotné aplikace.

2.3.6 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Pro samotnou kompilaci je nutné změnit typ daného souboru, nebo vytvořit nový soubor podporovaného typu (např. m), překopírovat text do tohoto souboru a následně ho zkompilovat pomocí jednoho ze tří zmíněných programů.



Obrázek 10: Ukázka editoru Etherpad

3 Rešerše zdarma poskytovaných nástrojů (Web aplikace)

3.1 Overleaf

Původně se jednalo o editor s názvem ShareLaTeX, který fungoval jako aplikace, která byla provozována na serveru a byla přístupná prostřednictvím webového prohlížeče. V červenci roku 2017 byl získán firmou Overleaf. ShareLaTeX byl upraven a od září 2018 je prezentován jako Overleaf v2. Oficiální webová stránka ShareLaTeX slouží pro přesměrování na Overleaf v2. Overleaf poskytuje pohodlí snadno použitelného editoru LaTeX se spoluprací v reálném čase a plně kompilovaný výstup vytvořený automaticky na pozadí během psaní.[14]

Aplikaci lze najít na této webové adrese:

<https://cs.overleaf.com/>

3.1.1 Požadavky

- Připojení k Internetu
- Webový prohlížeč
- Je zde rozdělení do 3 verzí editoru:
 - Personal (zdarma)
 - Collaborator (placené)
 - Professional (placené)

3.1.2 Funkčnost

Funkčnost editoru lze rozdělit podle verzí jak ukazuje následující tabulka:

Tabulka 2: Přehled Overleaf funkcí [15]

	Personal	Collaborator	Professional
Cena	Zdarma	14 eur/měsíc	28 eur/měsíc
Počet spolupracovníků	1	10	neomezeno
Neomezený počet privátních projektů	ano	ano	ano
Kolaborace v reálném čase	ano	ano	ano
Šablony	ano	ano	ano
LaTeX editor	ano	ano	ano
Sledování změn v reálném čase	ne	ano	ano
Pokročilé hledání odkazů	ne	ano	ano
Synchronizace manažera referencí	ne	ano	ano
Historie celého dokumentu	ne	ano	ano
Dropbox integrace	ne	ano	ano
GitHub integrace	ne	ano	ano
Prioritní podpora	ne	ano	ano

Mezi další funkce patří automatické zobrazení v reálném čase, což po kompilaci na pozadí má za následek zobrazení výsledného PDF souboru. Dále umožňuje komentování v reálném čase a chat mezi spolupracovníky. Editor umožňuje práci ve dvou režimech: Rich text a LaTeX. Také obsahuje rychlé vyhledání chyb a upozornění v kódu.

3.1.3 Propojení

Každý dokument, který vytvoříte na Overleaf, je ve výchozím nastavení soukromý se dvěma snadnými způsoby, jak sdílet svou práci se spolupracovníky:

- Soukromé pozvánky - umožňují pozvat vybrané, pojmenované spolupracovníky k přístupu k projektům. Počet jmenovaných spolupracovníků, které lze pozvat na každý projekt, závisí na verzi editoru.
- Sdílení odkazů - umožňuje sdílet vaše projekty prostřednictvím tajných odkazů. Stačí zapnout sdílení odkazů, odeslat odkaz vašim spoluautorům a mohou si je prohlížet, komentovat a upravovat.

[32]

3.1.4 Instalace

Instalace není potřeba, aplikace plně funguje ve webovém rozhraní.

3.1.5 Licence

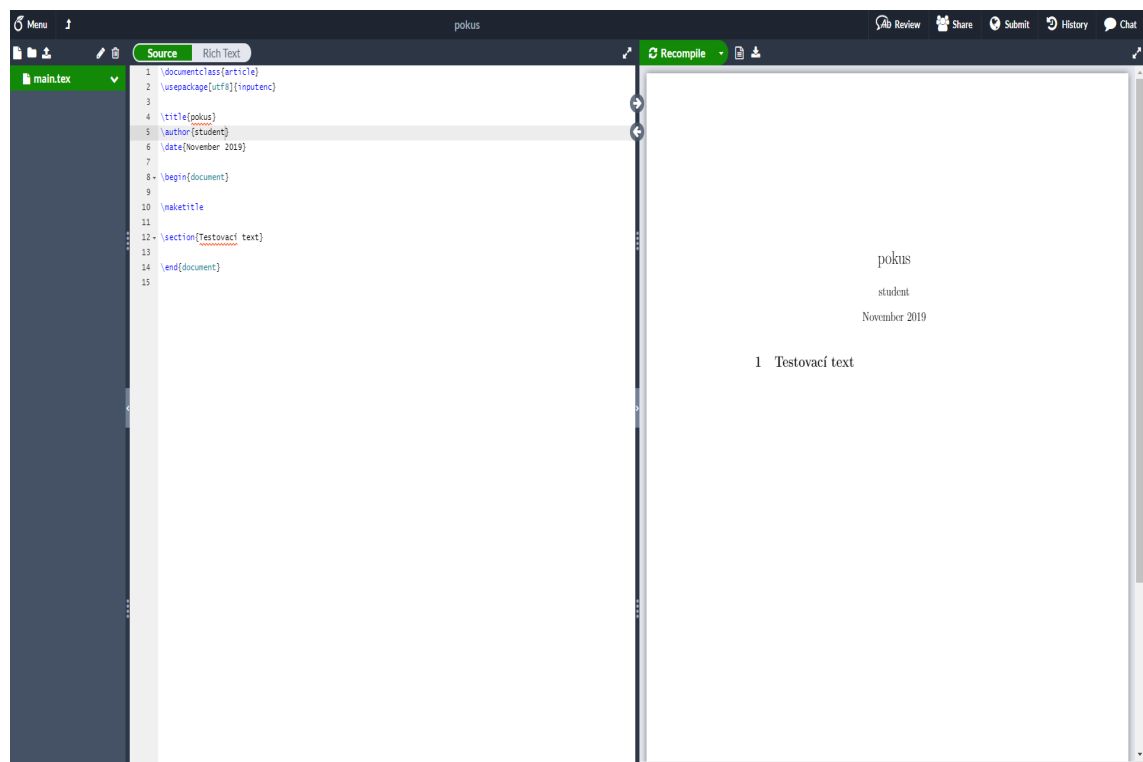
- ShareLaTeX - původně vedený pod licencí GNU Affero General Public License
- Overleaf v2 - pro osobní užití zdarma, jinak placené tarifem

3.1.6 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Verze Collaborator a Professional poskytují přímou integraci s verzovacím systémem Git, GitHub a Dropbox.

3.1.7 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Vzhledem k tomu, že se jedná o LaTeX editor, tak zde není přímá možnost ke kompilaci. Jazyk jako takový není danými prostředími zpracovatelný.



Obrázek 11: Ukázka editoru Overleaf

3.2 CryptPad

Jedná se o alternativu k oblíbeným kancelářským nástrojům a cloudovým službám. Editor nabízí poměrně širokou škálu možností k editaci zdrojových kódů a textů. Webová stránka CryptPad.fr je oficiální instancí projektu CryptPad s otevřeným zdrojovým kódem. Spravuje ji XWiki SAS, francouzská společnost vlastněná zaměstnanci, která tento produkt vytvořila a i nadále udržuje. Lze velice rychle vytvářet dokumenty pro spolupráci, které také umožňují psaní poznámek a připomínek.[16] Hlavní aplikace samotného CryptPadu jsou pak:

- CryptDrive - datové úložiště
- Rich text - pro editaci Rich textu
- Code - pro editaci kódů (zejména html)
- Presentation - pro tvorbu prezentací
- Poll - pro vytváření dotazníků ohledně možností k určitému tématu
- Kanban - diář pro vytváření úkolů, zaznamenávání a sledování jejich postupů
- Whiteboard - pro sdílené kreslení

3.2.1 Požadavky

- Připojení k Internetu
- Webový prohlížeč
- Dále je zde rozdělení do 3 kategorií uživatelů, kde každá kategorie nabízí jiný přístup k funkcím:
 - Anonymous user (zdarma)
 - Registered user (zdarma)
 - Premium user (zpoplatněno měsíčně)

3.2.2 Funkčnost

Jednotlivé aplikace poskytují do jisté míry vlastní funkcionalitu danou charakterem aplikace. V rámci této práce si probereme funkce aplikace Code. Po vstupu do samotného Code editoru lze spatřit rozdělení pracovní plochy na dvě části. Levá část slouží pro samotné psaní kódu a pravá poskytuje zobrazení Markdown⁴ náhledu (záleží na nastavení konkrétního programovacího jazyku). V levém horním rohu lze spatřit velký symbol CryptPadu, který uživatele přesměruje na jeho CryptDrive (cloudové úložiště). Lze také změnit název dokumentu, sledovat počet účastníků

⁴Markdown je jednoduchý značkovací jazyk se syntaxí formátování prostého textu.[33]

(rozděleno na ty, co mají právo číst, nebo s možností i dokument editovat). Dále pak je zde možnost chatu a posílání pozvánek prostřednictvím odkazu ke sdílení. Editor také nabízí možnost vybrat si z obrovské nabídky jazyků (i programovacích) a samozřejmě i import a export kódu. Bohužel zde není možnost přímo ovlivnit cílové uložení souboru při exportu (nastaveno podle prohlížeče), pojmenování souboru nelze provést při samotném exportu, ale jak bylo zmíněno výše, tak lze při samotné editaci. Typ exportovaného souboru se adaptuje podle použitého jazyka.

Lze také upravit přezdívku uživatele, další nastavení ohledně účtů a cloudového úložiště. Prohlédnout si historii úprav, importovat soubory a tisknout dokumenty.

3.2.3 Propojení

K propojení dochází pomocí výše zmíněných odkazu pro sdílení, u kterého lze nastavit, jestli bude daný dokument určen k úpravě či pouze pro čtení. CryptPad používá (podle slov tvůrců) stoprocentní šifrování na straně klienta. Po registraci a přihlášení je uživatelské jméno a heslo použito k vypočtení privátního klíče. Tyto položky nejsou nikdy posílány na server. Místo toho jsou použity k dešifrování obsahu uloženého v CryptDrivu, který obsahuje klíče k všem dokumentům, ke kterým má dotčený uživatel přístup. Vše se odehrává na klientské straně. Při sdílení odkazu na dokument, je sdílen kryptografický klíč pro přístup k tomuto dokumentu, ale protože je klíč uložen v tzv. identifikátoru fragmentu, tak není nikdy odeslán přímo na server.[16]

3.2.4 Instalace

Samotné aplikace editoru fungují ve webovém prohlížeči na webové stránce:

<https://cryptpad.fr/>

Instalace proto není nutná, ale je zde i možnost provést instalaci na Linuxu. Vzhledem k povaze zdroje, kde autor přímo uvádí, že je možné editovat a přidávat chybějící věci k tomuto návodu, jsem se rozhodl zde pouze uvést odkaz na daný návod:

<https://github.com/xwiki-labs/cryptpad/wiki/Installation-guide>

3.2.5 Licence

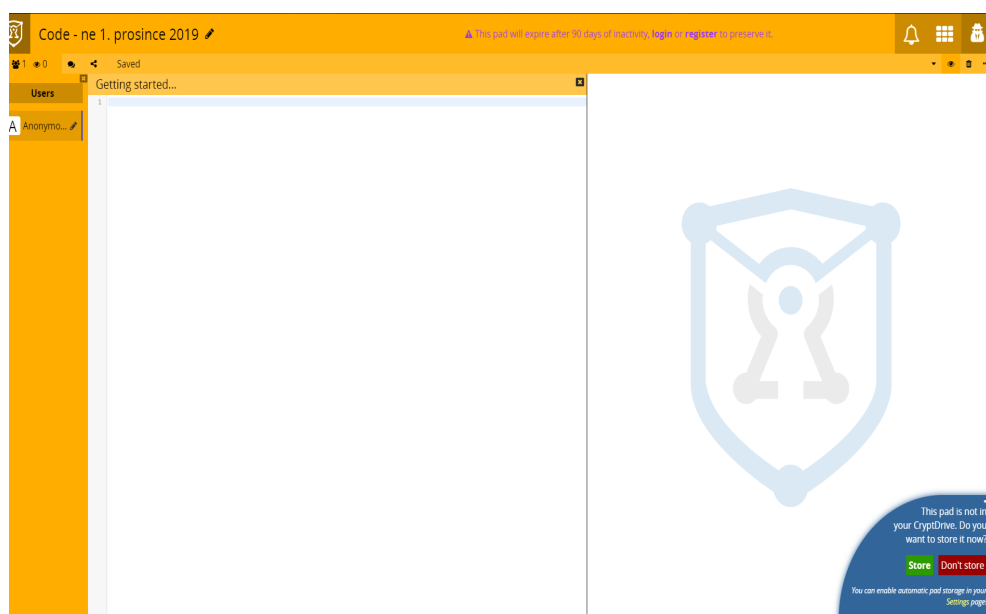
Tento editor je vedený pod licencí GNU Affero General Public License

3.2.6 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Editor neumožňuje propojení se systémy pro správu verzí prostřednictvím samotné aplikace. Editor používá vlastní řešení úložiště.

3.2.7 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Vzhledem k rozsáhlé možnosti výběru daného jazyka lze říci, že lze exportovat soubory daného typu, která tato prostředí umí zpracovávat.



Obrázek 12: Ukázka editoru CryptPad

3.3 Meetingwords

MeetingWords je jednoduchý textový editor, fungující ve webovém prohlížeči. Byl vytvořen Peterem Kaminským s podporou dobrovolníků. Editor používá EtherPad Open Source engine, který byl vytvořen softwarovými inženýry z firmy AppJet, kterou později koupila firma Google.[17]

3.3.1 Požadavky

- Připojení k Internetu
- Webový prohlížeč

3.3.2 Funkčnost

Editované soubory jsou uloženy na webu, což k nim umožňuje přistupovat z libovolného počítače. Lze také přizvat ostatní spolupracovníky (maximálně však 32 v jeden okamžik) k editaci dokumentu v reálném čase. Editor není příliš vhodný pro dlouhodobější ukládání dokumentů, protože tzv. Pady jsou udržovány maximálně 7 dní.[17]

- Editor umožňuje základní editaci textu
- rozlišení spolupracovníků pomocí barev
- očíslování řádků
- import a export ve formátech plain text, html, bookmark file a jsou zde i další možnosti, bohužel zneprístupněny
- možnost chatu a úpravy jména každého jednotlivého uživatele
- ukládání a prohlížení historie úprav

3.3.3 Propojení

Sdílení funguje na principu vygenerování a sdílení odkazu pro sdílení mezi spolupracovníky. Také lze odeslat pozvánku ke spolupráci pomocí emailu.

3.3.4 Instalace

Editor není třeba instalovat. Funguje jako webová aplikace, která je připravená k okamžitému užívání.

3.3.5 Licence

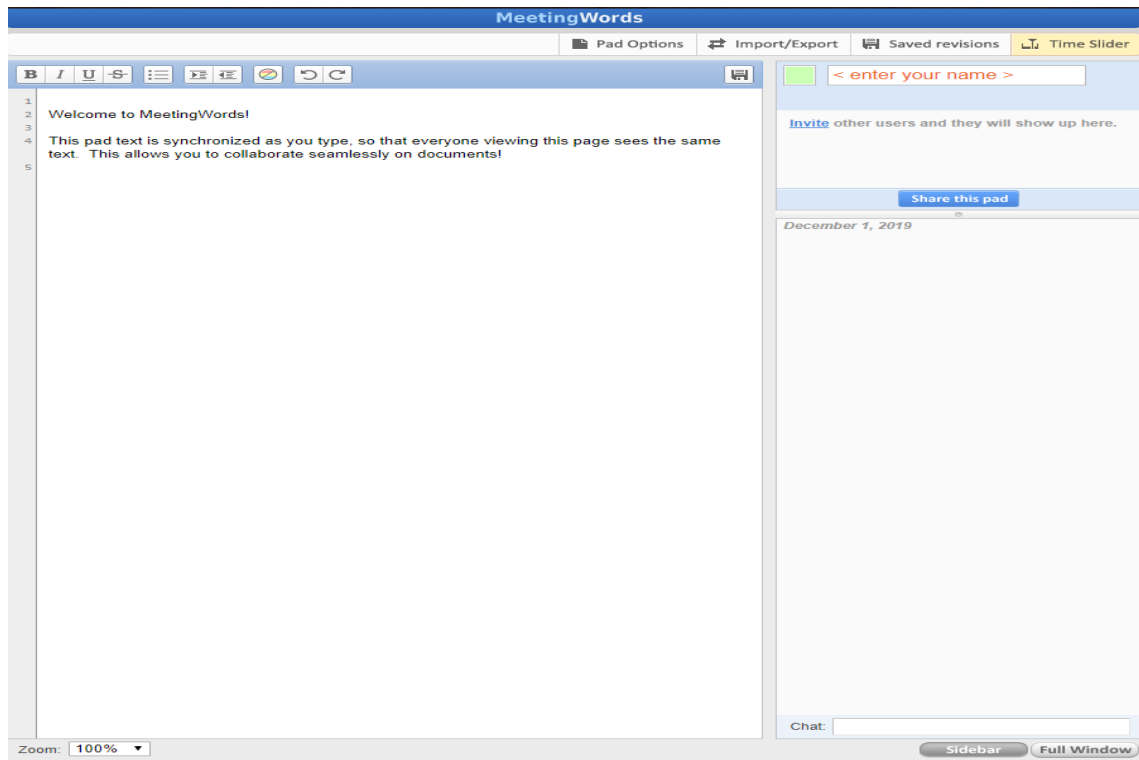
MeetingWords funguje jako řešení zdarma, ale je možno autorům zaslat příspěvek formou tzv. donations, což je zcela dobrovolné a takto zaslané příspěvky slouží převážně k pokrytí nákladů za provoz serverů, na kterých je aplikace hostována.[17]

3.3.6 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Editor neumožňuje propojení se systémy pro správu verzí prostřednictvím samotné aplikace.

3.3.7 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Editor nepodporuje export příslušných formátů a možnost výběru uložení, či přejmenování samotného souboru. Nutno provést ručně pomocí vytvoření nového souboru určeného pro kompilaci a přepokopírovat text, či kód do tohoto souboru.



Obrázek 13: Ukázka editoru MeetingWords

3.4 FirePad

Jedná se o Open Source real-time kolaborativní textový editor. Poskytuje, kromě již zmíněné možnosti kolaborace, také inteligentní sloučení založené na Operační transformaci⁵ a řešení konfliktů. Např. synchronizaci pozice kurzoru, Undo/Rendo, Text Highlighting, uživatelské atributy, detekce přítomnosti uživatele a ukládání verzí. Užití FirePadu je velice specifické, byl navržen, aby fungoval jako vestavěná aplikace uvnitř větších aplikací. Používá Firebase realtime databázi jako backend. Tím pádem nevyžaduje běh kódu na straně serveru a může být přidán do jakékoli webové aplikace jednoduchou implementací JavaScript souborů.[19]

3.4.1 Požadavky

- soubory JavaScript, které je nutno zahrnout do konkrétní webové stránky, či aplikace
- účet pro Firebase

3.4.2 Funkčnost

Většina textových editorů pro spolupráci vyžaduje, aby na serveru fungoval speciální kód, což je nepraktické, pokud již nepoužíváme správný serverový zásobník⁶. Firepad nemá závislost na serveru a místo toho spoléhá na synchronizaci dat v reálném čase na databázi Firebase Realtime Database. To znamená, že je snadné přidat Firepad do jakékoli aplikace, dokonce i do statických webů.[19]

3.4.3 Propojení

Firebase Realtime Database je cloudová databáze. Data jsou ukládána jako JSON a synchronizována v reálném čase s každým připojeným klientem. Při vytváření aplikací napříč platformami se sadami pro iOS, Android a JavaScript SDK sdílí všichni klienti jednu instanci databáze Realtime a automaticky dostávají aktualizace s nejnovějšími daty.[21]

Konkrétní způsob přizvání spolupracovníků je však na každém, kdo se rozhodne k implementaci své aplikace využít FirePad.

3.4.4 Instalace

Instalace FirePadu je dosti specifická, protože se nejedná o klasické řešení formou desktopové, či samostatné webové aplikace. Chcete-li vložit Firepad do své vlastní aplikace, musíte se nejprve zaregistrovat k bezplatnému účtu Firebase na následující stránce:

<https://firebase.google.com/>

⁵Operační transformace (OT) je technologie pro podporu celé řady funkcí spolupráce v pokročilých kolaborativních softwarových systémech.[18]

⁶Zásobník serveru je kolekce softwaru, který tvoří operační infrastrukturu na daném počítači.[22]

Zájemci o vytvoření nového projektu Firebase a samotné konfigurace můžou tak učinit podle FirePad dokumentace na následující stránce:

<https://firepad.io/docs/>

Důležité je také zmínit, že konfigurace se bude lišit v případě, že si budete přát využít FirePad pro editaci tzv. Rich Text souboru. V takovém případě je nutné zvolit konfiguraci s Codemirror. Pro ostatní případy lze využít jak Codemirror, tak Ace konfiguraci.

[20]

3.4.5 Licence

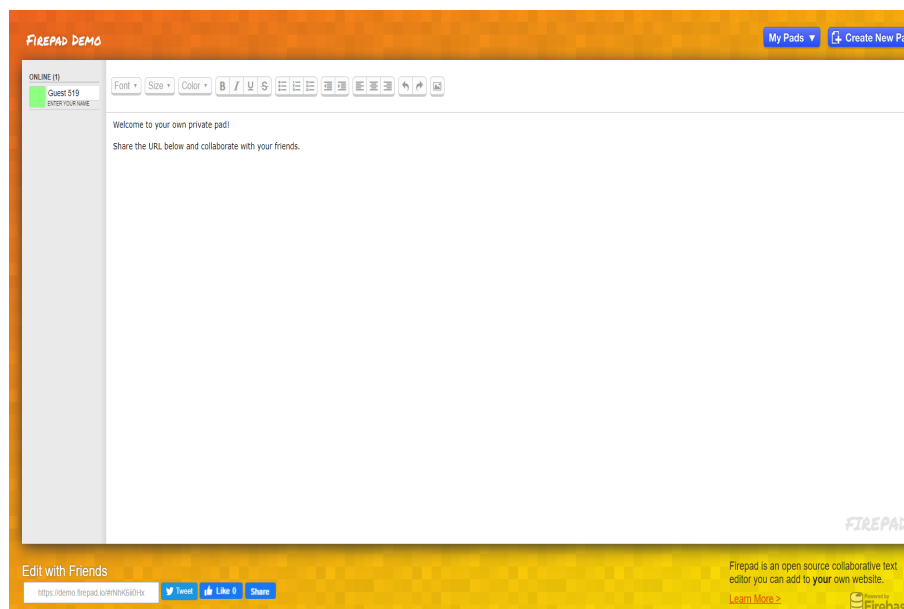
FirePad je licencovaný licencí MIT, kterou lze kombinovat s GPL softwarem.[23]

3.4.6 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

Editor neumožňuje propojení se systémy pro správu verzí prostřednictvím samotné aplikace.

3.4.7 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Editor nepodporuje export formátů a možnost výběru uložení, či přejmenování samotného souboru. Nutno provést ručně pomocí vytvoření nového souboru určeného pro kompilaci a přepírat text, či kód do tohoto souboru.



Obrázek 14: Ukázka editoru Firepad

3.5 Conclave

Conclave je peer-to-peer, kolaborativní textový editor pro editaci v reálném čase, který byl vytvořen pomocí JavaScriptu. Editor funguje v prostředí internetového prohlížeče. Pro synchronizaci používá Conflict-Free Replicated Data Type (CRDT)⁷ a WebRTC⁸ pro posílání přímých zpráv ostatním účastníkům.[24]

3.5.1 Současnost

V současné době Conclave už není podporováno a dále vyvíjeno. Veřejně přístupný editor funguje jako demo verze, která umožňuje základní kolaboraci ve smyslu společného psaní textu. Chybí zde možnosti jako nastavení velikosti písma, fontu, zarovnání a další. Zvýraznění syntaxe také není podporováno. Lze ovšem nahrávat obyčejné textové soubory (např. txt) a po úpravě si lze soubor z dané relace stáhnout na lokální počítač. Demo editoru neumožňuje stažený soubor pojmenovat a určit místo uložení na lokálním disku. Editor sám pojmenuje a uloží data jako obyčejný textový soubor (např. txt) ve formě planého textu. Ukládání na serveru není možné. Sdílení dokumentu je možné prostřednictvím náhodně vygenerovaného odkazu na konkrétní relaci toho, kdo tento odkaz vygeneroval a následně sdílel s ostatními spolupracovníky. Demo verzi lze nalézt na následujícím odkazu:

<https://www.conclave.tech/>

3.5.2 Možnost propojení daného nástroje s verzovacím systémem Git a GitHubem

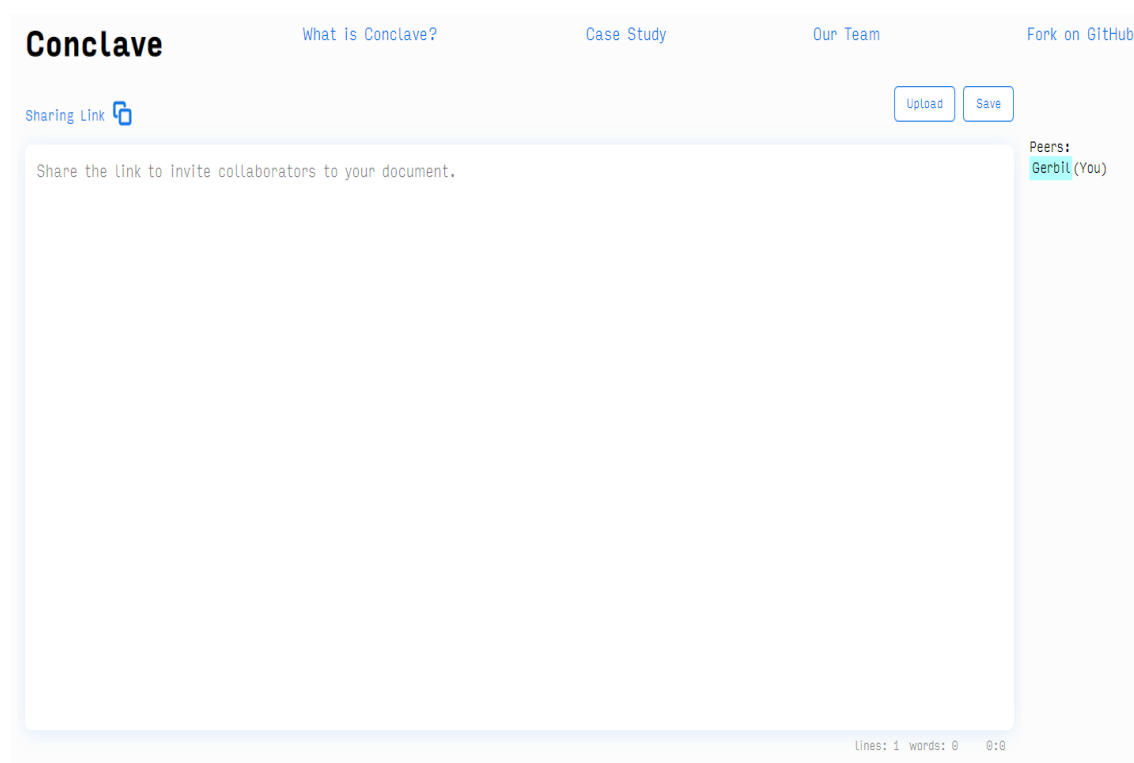
Jelikož se jedná o demo verzi editoru, která už dávno není podporována, tak bohužel tuto možnost zde nelze očekávat.

3.5.3 Možnosti lokální kompilace v prostředích MATLAB, Scilab a GNU Octave

Pro samotnou kompilaci je nutné změnit typ daného textového souboru, nebo vytvořit nový soubor podporovaného typu (např. m), překopírovat text to tohoto souboru a následně ho zkompileovat pomocí jednoho ze tří zmíněných programů.

⁷Zjednodušeně řečeno se tyto datové typy chovají autonomně a stále zajišťují konzistenci. [37]

⁸Jedná se o specifikaci HTML5, kterou lze použít k přidání mediální komunikace v reálném čase přímo mezi prohlížeč a zařízení. [36]



Obrázek 15: Ukázka editoru Conclave

4 Popis a možnosti využití dalších nástrojů pro výuku

4.1 Octave online

GNU Octave je vysokoúrovňový interpretovaný jazyk, určený především pro numerické výpočty. Poskytuje schopnosti pro numerické řešení lineárních a nelineárních problémů a pro provádění dalších numerických experimentů. Poskytuje také rozsáhlé grafické funkce pro vizualizaci a manipulaci s daty. GNU Octave se obvykle používá prostřednictvím svého interaktivního rozhraní (CLI a GUI), ale lze jej také použít k psaní neinteraktivních programů. Jazyk GNU Octave je do značné míry kompatibilní s Matlabem, takže většina programů je snadno přenositelná. Kromě toho jsou podporovány funkce známé ze standardní knihovny C a ze systémových volání a funkcí systému UNIX. C, C++ a Fortran kód lze volat z Octave vytvořením Oct-Files nebo pomocí Mat-File kompatibilních s Matlabem.[25]

Octave Online bere tento koncept a přesouvá ho z lokálního prostředí počítače do webového a umožňuje tedy využití jako webové služby s možností propojení ostatních přispěvatelů.

Aplikaci lze najít na této webové adrese:

<https://octave-online.net/>

4.1.1 Požadavky

- Připojení k Internetu
- Webový prohlížeč
- Pro sdílení dokumentu je potřeba přihlášení pomocí Google+ účtu nebo emailu.

4.1.2 Instalace

Instalace není potřeba, aplikace plně funguje ve webovém rozhraní.

4.1.3 Licence

Tento editor je vedený pod licencí GNU Affero General Public License

4.1.4 Funkčnost a využití pro možnosti výuky

Jak už bylo zmíněno, jedná se o webovou aplikaci. Octave Online nabízí provoz ve dvou módech:

- **Mód bez sdílení**

V podstatě tento mód funguje jako klasický příkazový řádek GNU Octave. Lze do něj psát proměnné, funkce atd. Výhodou je, že uživateli stačí pouze připojení k internetu bez nutnosti mít GNU Octave nainstalován na svém fyzickém počítači. Nevýhodou je bohužel

to, že bez přihlášení není možné dané soubory sdílet, případně jednotlivé verze ukládat online.

- **Mód se sdílením**

Tento mód také poskytuje standardní funkce příkazové řádky GNU Octave, ale navíc umožňuje sdílení pracovního souboru s ostatními přispěvateli prostřednictvím odkazu ke sdílení a také ukládání verzí souboru pomocí Gitu. K souborům lze přistupovat online, pomocí přihlašovacích údajů, nebo je lze stáhnout a spravovat lokálně. Tento mód také podporuje možnosti tisku.

Toto řešení by mohlo mít pozitivní přínos pro výuku, protože umožňuje studentům pomocí jediné registrace okamžitě začít pracovat v prostředí Octave, aniž by se museli seznamovat se složitějšími funkcemi a rozložením grafického GNU Octave IDE, nebo pracovali čistě v klasické příkazové řádce na lokálním počítači. Co si myslím, že není úplně vhodné je to, že vývojáři, kteří tento projekt vytvořili a nadále poskytují podporu, uvádí ve svých podmínkách jaká data ohledně uživatel sbírají. Především se jedná o samotné skripty a kontaktní informace ve formě emailu nebo Google+ účtu.[26]

**Více o sběru těchto informací a licenčních podmínkách se lze dočíst v záložce
Privacy Policy and EULA v samotné aplikaci.**

4.2 Scilab Cloud API

Scilab Cloud API je webová služba, která poskytuje přístup ke sdíleným souborům, které jsou dostupné na jakémkoli počítači. Uživatel si vytvoří nějaké výpočetní, případně jiné funkce, které se rozhodne používat a sdílet prostřednictvím Scilab Cloud API. Po nahrání funkcí může koncový uživatel tuto webovou službu volat pouze prostřednictvím HTTP žádosti, díky které získá výsledky.[27]

Webovou službu s funkcí lze potom volat aplikacemi třetích stran jako například:

- Scilab desktop klient
- Jednoduchá webová stránka s grafikou Javascriptu
- Google spreadsheet

Zde je odkaz na oficiální video, jak webové služby volat:

https://www.youtube.com/watch?time_continue=240&v=9KCLmFaQfmY&feature=emb_title

Podporovaná prostředí jsou Windows, Linux a Mac.

4.2.1 Požadavky

- mít připravené funkce pro nahrání
- „zabalit funkce“ to tzv. Toolboxu
- Pro přístup do Scilab Cloud administrátorského interface je potřeba kontaktovat Scilab team dát odběr na účet Scilab Cloud: team@scilab.io (email)

[28]

4.2.2 Další informace

Pro detailní postup a informace doporučuji prostudovat uživatelský manuál:

<https://www.scilab.org/cloud/web-services>

4.2.3 Funkčnost a využití pro možnosti výuky

Aplikace, respektive služba má celkem velký potenciál při výuce, kde by mohla fungovat jako databáze pro shromažďování Scilab funkcí, které jsou využívány jak pro výuku, tak i pro výzkumnou práci. Nevýhodou je to, že princip „zabalení“ funkcí do Toolbox balíčku před nasazením do služby je poněkud složitější.

4.3 Matlab drive

MATLAB Drive je software, který umožňuje kolaboraci, ukládání a synchronizaci MATLAB souborů na bázi cloudového úložiště. Pro držitele placené licence poskytuje až 5 GB úložného prostoru, případně 250 MB úložného prostoru zdarma pro uživatele, kteří si založí účet MathWorks. Úložiště funguje s aplikacemi:

- MATLAB Drive online
- MATLAB Mobile
- MATLAB Online

Podporovaná prostředí jsou: MS Windows, Linux a Mac.

[29]

Aplikaci lze najít na této webové adrese:

<https://www.mathworks.com/products/matlab-drive.html#explore-matlab-drive>

4.3.1 Požadavky

- MathWorks účet
- MATLAB Connector (desktopové řešení)
- Podporovanou verzi webového prohlížeče

4.3.2 Instalace

Instalace je potřebná jenom v případě desktopového řešení MATLAB Connector, přičemž stačí stáhnout instalační program odpovídající operačnímu systému a nainstalovat. Po uživateli je také požadován MathWorks účet, který lze založit zdarma.[29]

4.3.3 Licence

Placená i zdarma s omezeným úložištěm

4.3.4 Funkčnost a využití pro možnosti výuky

Aplikace umožňuje sdílení a kolaboraci tak, že ostatní pozvete do složky, kterou si přejete sdílet a můžete pro ostatní nastavit práva jestli můžou editovat nebo si jenom čistě prohlížet vaši práci. Fungují zde jistá omezení podle licencí s kolika přispěvateli můžete dané složky sdílet. Sdílení probíhá formou pozvánky, která je dotyčné osobě zaslána na email, případně vytvořením sdíleného odkazu pouze pro prohlížení.

Více informací ohledně funkcí sdílení a další lze nalézt v následující dokumentaci:
<https://www.mathworks.com/help/matlabdrive/ug/share-and-collaborate-on-matlab-drive.html>

Využití řešení je vcelku velké, zejména pro studenty, kteří využívají MATLAB. Vzhledem k rozsáhlosti dokumentace a funkčnosti může chvíli trvat, než se uživatel zorientuje, ale nejedná se o nic významně složitějšího. Hlavní výhodou je to, že vše je hezky provázané a sjednocené na jednom místě, jak z hlediska úložiště tak i možností sdílení pro kolaboraci. Dokonce pro využívání tohoto řešení není potřeba samotný MATLAB, což přidává na flexibilitě. Nevýhodou může být omezení kvůli různým druhům licencí.

5 Porovnání a výběr softwaru pro výuku ze zdarma nabízených řešení

K porovnání jsem zmenšil výběr zde zmíněných zdarma poskytovaných editorů pouze na dva: Atom a Gobby. Vybral jsem je z toho důvodu, že se jedná o samostatné desktopové aplikace, které neběží někde na webu v omezené verzi a dají se dokonce i rozšířit a konfigurovat.

V následující tabulce lze vidět porovnání základních vlastností editorů.

Tabulka 3: Porovnání vlastností editorů

Vlastnost editoru	Atom	Gobby
Desktopová aplikace	Ano	Ano
Rozšíření, konfigurovatelnost	Ano	Ano
Intuitivní používání	Ano	Ano
Je aplikace bezpečná?	Ano (do určité míry)	Ano
Podpora zvýrazňování syntaxí jazyků	Ano	Ano
Přímé propojení s verzovacími systémy	Ano	Ne
Sdílení hned po instalaci	Ne	Ano
Provoz vlastního serveru	Ne	Ano
Přímé sdílení bez externí komunikace?	Ne	Ano

- **Proč zrovna desktopovou aplikaci?**

Výhodou samotné desktopové aplikace je bezpochyby to, že SW jako takový je dodáván jako celistvý balíček (pokud se vysloveně nejedná o demo verzi), který by už ve svém základním stavu měl poskytovat dostatečnou funkcionalitu pro práci. Což většina webových editorů, které jsou zmíněny v rešerši nesplňuje. Důkazem toho může být příklad, že při obyčejné potřebě uložení souboru na lokální systém není umožněno pojmenování, ani výběr místa uložení, což lze považovat za jednu z mnoha základních funkcionalit.

Atom a Gobby toto kritérium však splňují.

- **Rozšiřitelnost a konfigurovatelnost**

V případě Atomu se touto vlastností vyznačuje především modifikace a rozšíření samotného editoru, jako příklad bych zde mohl uvést, že Atom v základní instalaci není schopen kolaborace, je tedy nutné editor rozšířit o plugin (balíček) Teletype. Taková forma rozšiřitelnosti je vcelku jednoduchá a příjemná pro uživatele, protože mu umožňuje si svůj editor přizpůsobit podle svých potřeb.

V případě Gobby editor jako takový rozšířit formou pluginů nelze, ale server do jisté míry ano, což zase napomáhá správci při zabezpečení dat a komunikace.

- **Intuitivnost**

V případě Atomu je editor velice uživatelsky přívětivý a není obtížné se v něm po chvilce zorientovat. Základní menu je jednoduché, instalace balíčků probíhá prostřednictvím online průzkumníku, do kterého napíšeme název balíčku, který si přejeme nainstalovat, ten poté stáhneme a nainstalujeme. Sdílení a změna syntaxe jazyka je také jednoduchá formou kliknutí na ikonu a výběru potřebného jazyka.

V případě Gobby může uživateli z počátku dělat problémy se zorientovat v hledání potřebných funkcí, protože jsou většinou skryté pod různými sekcemi v menu, ale po chvilce užívání by už uživatel neměl mít problém editor používat. Sdílení lze uskutečnit buďto přes přímé zadání IP adresy serveru nebo klienta v levém dolním menu, nebo si vybrat z Gobby propagovaných uzlů, které se zobrazují v okně Document Browser. V menu Edit, v záložce Preferences, lze měnit základní nastavení editoru, jako například heslo pro vzdálené uživatele, kteří by se rádi připojili na Váš počítač formou přímého spojení. Případně pak další jako změna barevného rozlišení uživatele v editoru, uživatelské jméno atd.

- **Bezpečnost**

S vylepšováním a technologickým pokrokem, který přináší nové možnosti komunikace, vzniká také čím dál větší potřeba a důraz na bezpečnost. U sdílení tomu určitě není jinak.

V případě Atomu navázání spojení a sdílení probíhá tak, že nejdříve editor kontaktuje externí server pro navázání spojení s jiným editorem. Komunikace pak probíhá čistě mezi

klientskými systémy a je šifrována WebRTC. Vývojáři tvrdí, že zde není žádný centralizovaný server, který by odposlouchával sdílenou komunikaci. Otázkou je, jestli je to skutečně pravda? Navíc v případě výpadku, nebo problému s komunikací se servery, nemusí ke spojení mezi editory vůbec dojít.

Gobby v tomto ohledu nabízí jiné řešení. Ke komunikaci lze buďto využít vlastního serveru, který lze nakonfigurovat podle potřeby, nebo se lze ihned připojit přímo ke klientovi, pokud známe jeho IP adresu (případně i heslo). Navázání komunikace zde nemusí být vedeno přes externí servery. Tím lze zajistit, že vše bude probíhat v rámci naší sítě.

- **Podpora zvýraznění syntaxe**

Z hlediska komfortu práce je rozlišování klíčových slov, příkazů, atd. určitě velikým ulehčením práce a zlepšením přehlednosti. Jak Atom, tak Gobby nabízejí vcelku široký výběr mezi zvýrazněním syntaxí u různých programovacích jazyků. Tuto vlastnost bychom u většiny výše zmíněných webových editorů hledali marně.

- **Propojení s verzovacím systémem** Atom umožňuje přímou komunikaci s verzovacím systémem díky integrované funkci. Stačí jenom vybrat v pravém dolním rohu editoru (Git, Github) a publikovat.

Gobby naproti tomu podobnou funkci nenabízí a je nutné tuto funkčnost řešit ručním publikováním souborů přes příkazovou řádku, případně jiné rozhraní.

- **Sdílení**

Atom ihned po instalaci není schopen poskytovat sdílení souborů ke kolaboraci. Proto je nutné stáhnout a nainstalovat rozšiřující balíček Teletype.

Gobby ihned po instalaci umožňuje sdílení na přímo mezi klientskými systémy, nebo přes komunikační server. Není nutné stahovat a instalovat další rozšíření.

- **Provoz vlastního serveru pro komunikaci**

Atom tuto vlastnost nepodporuje, komunikaci mezi editory je nutné navázat přes externí servery, které nemůžeme mít ve správě.

Gobby nabízí možnost provozu vlastního konfigurovatelného serveru, jehož základní funkcionality lze rozšířit formou pluginů.

- **Přímé sdílení bez nutnosti externí komunikace** Sdílení v Atomu funguje na principu přímé komunikace mezi klienty, ale až poté, co se toto spojení sestaví za přítomnosti externího serveru jako prostředníka, což při problémech s externí sítí nebo nefunkčnosti serveru může způsobit, že k propojení nedojde.

Gobby naproti tomu, jak už jsem zmínil několikrát, umožňuje v tomto možnost volby přímého sdílení bez nutnosti provozovat server, nebo sdílení za přítomnosti serveru může být součástí a pod správou naší sítě.

Závěr

Hlavním cílem této práce bylo provést průzkum volně dostupných řešení pro kolaborativní editaci kódu v reálném čase, s následným potencionálním využitím ve výuce. Jak se ukázalo, tak takových řešení je opravdu nepřehledné množství. Při mém výběru relativně malého množství, čítajícího osm editorů, jsem zjistil, že většina jsou v podstatě kopie založené na základech jednoho editoru, bohužel i se stejnými nedostatky, případně horší funkcionalitou. Jedná se především o webové editory u kterých se neupřednostňuje kvalita, ale spíš kvantita, kvůli snadné dostupnosti apod.

Naproti tomu jsem také našel velice kvalitní editory, které si myslím, že pro účely výuky budou více než dostatečné. Těmito editory jsou, jak už jsem uvedl v předchozí kapitole, Atom a Gobby s tím, že Atom vzhledem ke své obrovské možnosti vlastní konfigurace a přizpůsobení si editoru k vlastním potřebám je o kousek lepší než Gobby, který zase vyniká umožněním celkové kontroly nad správou vlastního serveru a zároveň se jedná o bezpečnější variantu, kdy se nepoužívají externí servery pro navázání komunikace mezi uživateli.

Tuto práci jsem také částečně pojal jako návod a lehký úvod pro ty čtenáře, kteří se nikdy nesetkali s verzovacím systémem a chtěli by ho začít používat a také, jak zmíněné editory instalovat, konfigurovat a použít, včetně příkazů pro instalaci a konfiguraci, které je možno kopírovat a přímo je použít, což si myslím, že čtenáři ocení.

Literatura

- [1] Git: About. CHACON, Scott, Jason LONG a Contributors. SOFTWARE FREEDOM CONSERVANCY. Git [online]. New York, 2012, 4. 5. 2012 [cit. 2020-04-26]. Dostupné z: <https://git-scm.com/about>
- [2] BROWN, Korbin. What Is GitHub, and What Is It Used For? How-To Geek [online]. LifeSavvy Media, c2020, 13 November 2019 [cit. 2020-04-26]. Dostupné z: <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>
- [3] TUTS MAKE. Upload Project/Files On Github Using Command line. Tuts Make [online]. Tuts Make, c2020, 2. března 2019 [cit. 2020-04-24]. Dostupné z: <https://www.tutsmake.com/upload-project-files-on-github-using-command-line/>
- [4] CONTRIBUTORS, simurai a burodepeper. Atom: language-examples. GITHUB, INC. Github.com [online]. San Francisco (CA), c2020, 18. 9. 2015 [cit. 2020-04-25]. Dostupné z: <https://github.com/atom/language-examples>. README.md.
- [5] Electron. GITHUB, INC. Github.com [online]. San Francisco (CA), c2020, 21. 10. 2019 [cit. 2020-04-25]. Dostupné z: <https://github.com/electron/electron/blob/master/README.md>
- [6] Teletype for Atom [online]. 2017 [cit. 2020-04-25]. Dostupné z: <https://teletype.atom.io/>
- [7] BURGMEIER, Armin. Gobby. GITHUB, INC. Github.com [online]. San Francisco (CA), c2020, 27. 10. 2017 [cit. 2020-04-25]. Dostupné z: <https://github.com/gobby/gobby>. README.md Copyright (C) 2008-2015.
- [8] BURGMEIER, Armin. Infinoted-0.6 (1) - Linux Man Pages. <https://www.systutorials.com/> [online]. 2015, 11 Sep 2014 [cit. 2020-04-25]. Dostupné z: <https://www.systutorials.com/docs/linux/man/1-infinoted-0.6/>
- [9] CONTRIBUTORS, archlinux. Gobby. Archlinux [online]. archlinux, c2002-2020, 26 March 2013, 16 August 2019 [cit. 2020-04-25]. Dostupné z: <https://wiki.archlinux.org/index.php?title=Gobby&oldid=580054>
- [10] RODRIGUEZ « MAGICFAB », Fabian. GOBBY SERVER IN 3 STEPS. Fabian Rodriguez, « MagicFab » [online]. Montreal: MagicFab, 8. února 2010 [cit. 2020-04-24]. Dostupné z: <https://magicfab.ca/2010/02/gobby-server-in-3-steps/>
- [11] THE ETHERPAD FOUNDATION. Etherpad [online]. 2013, 24 April 2020 [cit. 2020-04-25]. Dostupné z: <https://etherpad.org/#about>

- [12] THE ETHERPAD FOUNDATION. Ether: etherpad-lite. GITHUB, INC. Github.com [online]. San Francisco (CA), c2020, 19. 4. 2020 [cit. 2020-04-25]. Dostupné z: <https://github.com/ether/etherpad-lite/blob/develop/README.md>. README.md.
- [13] KAHANAM, Essodjolo. Install Etherpad web-based real-time collaborative editor on Ubuntu 16.04 Linux. <https://linuxconfig.org/> [online]. linuxconfig.org, c2007-2020, 02 January 2017 [cit. 2020-04-24]. Dostupné z: <https://linuxconfig.org/install-etherpad-web-based-real-time-collaborative-editor-on-ubuntu-16-04-linux>
- [14] CONTRIBUTORS, Wikipedia. ShareLaTeX. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020, 22 avril 2020 [cit. 2020-04-25]. Dostupné z: <http://fr.wikipedia.org/w/index.php?title=ShareLaTeX&oldid=169886753>
- [15] Plans and Pricing. Overleaf [online]. c2020, 28 Sep 2018 [cit. 2020-04-25]. Dostupné z: <https://overleaf.com/user/subscription/plans>
- [16] What is CryptPad. CryptPad [online]. CryptPad, 2014, 18 Aug 2017 [cit. 2020-04-25]. Dostupné z: <https://cryptpad.fr/what-is-cryptpad.html>
- [17] KAMINSKI, Peter. Frequently Asked Questions. MeetingWords [online]. MeetingWords, c2010, 20 Apr 2010 [cit. 2020-04-25]. Dostupné z: <http://meetingwords.com/ep/about/faq>
- [18] CHENGZHENG, Sun. OT FAQ: Operational Transformation Frequently Asked Questions and Answers. NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE [online]. Singapur: NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE, c2020, c2010–2015 [cit. 2020-04-25]. Dostupné z: https://www3.ntu.edu.sg/home/czsun/projects/otfaq/#_Toc321146125. 1.1.What is OT?
- [19] FIREPAD Open source collaborative code and text editing [online]. FIREPAD, 2013 [cit. 2020-04-25]. Dostupné z: <https://firepad.io/#>
- [20] Documentation. FIREPAD Open source collaborative code and text editing [online]. FIREPAD, 2013, 9 Apr 2013 [cit. 2020-04-25]. Dostupné z: <https://firepad.io/docs/>
- [21] Firebase Realtime Database. Firebase [online]. Google, 2016, 8. 4. 2020 [cit. 2020-04-25]. Dostupné z: https://firebase.google.com/docs/database/?utm_source=landing&utm_medium=email&utm_campaign=firepad
- [22] ROUSE, Margaret. Server stack. WhatIs.com [online]. WhatIs.com, c1999-2020, July 2012 [cit. 2020-04-25]. Dostupné z: <https://whatis.techtarget.com/definition/server-stack>

- [23] GNU Operating System: Various Licenses and Comments about Them. GNU Operating System [online]. Boston (MA): Free Software Foundation, c1996-2020, 16. 3. 2020 [cit. 2020-04-25]. Dostupné z: <https://www.gnu.org/licenses/license-list.html.en#Expat>
- [24] SAVANT, Nitin, Elise OLIVARES a Sun-Li BEATTEAY. Conclave: A private and secure real-time collaborative text editor. Conclave Case Study [online]. conclave-team, 2018, 2018 [cit. 2020-04-25]. Dostupné z: <https://conclave-team.github.io/conclave-site/>
- [25] GNU Octave Wiki [online]. MediaWiki, 2011, 24 April 2020 [cit. 2020-04-25]. Dostupné z: https://wiki.octave.org/wiki/index.php?title=GNU_Octave_Wiki&oldid=12915
- [26] OctaveOnline [online]. 2013 [cit. 2020-04-25]. Dostupné z: <https://octave-online.net/>. Menu/Privacy Policy and EULA.
- [27] API: Web services. ESI GROUP. Scilab [online]. Rungis: Scilab, 2020, 29 Aug 2017 [cit. 2020-04-25]. Dostupné z: <https://www.scilab.org/cloud/web-services>
- [28] Scilab Cloud API: User Guide. Scilab [online]. Rungis: ESI Group, 2020, 29 Jun 2018 [cit. 2020-04-25]. Dostupné z: <https://www.scilab.org/sites/default/files/Scilab%20Cloud%20API%20-%20User%20manual.pdf>. Strana č. 3.
- [29] MATLAB Drive. THE MATHWORKS, INC. MathWorks [online]. Natick (MA), c1994-2020, 3 Dec 2016 [cit. 2020-04-25]. Dostupné z: <https://www.mathworks.com/products/matlab-drive.html>
- [30] CONTRIBUTORS, Wikipedia. Atom (text editor). In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020, 23.dubna 2020 [cit. 2020-04-24]. Dostupné z: [https://en.wikipedia.org/w/index.php?title=Atom_\(text_editor\)&oldid=952622117](https://en.wikipedia.org/w/index.php?title=Atom_(text_editor)&oldid=952622117)
- [31] Atom Flight Manual: Chapter 2: Using Atom - Atom Packages. Atom: A hackable text editor for the 21st Century [online]. 2018, 15. 3. 2016 [cit. 2020-04-27]. Dostupné z: <https://flight-manual.atom.io/using-atom/sections/atom-packages/>
- [32] Documentation: Sharing a project. Overleaf [online]. c2020, 19. 12. 2014 [cit. 2020-04-25]. Dostupné z: https://www.overleaf.com/learn/how-to/Sharing_a_project
- [33] Getting Started: What's Markdown? CONE, Matt. Markdown guide [online]. Albuquerque (NM), c2020, 20. 3. 2017 [cit. 2020-04-27]. Dostupné z: <https://www.markdownguide.org/getting-started/>
- [34] Cloning a repository. GITHUB, INC. GitHub Help [online]. San Francisco (CA), c2020, 10. 3. 2015 [cit. 2020-04-27]. Dostupné z: <https://help.github.com/en/github/creating-cloning-and-archiving-repositories/cloning-a-repository>

- [35] KELLEY, Kyle a Contributors. Atom-script: Script. GITHUB, INC. Github.com [online]. San Francisco (CA), c2020, 31. 10. 2019 [cit. 2020-05-07]. Dostupné z: <https://github.com/rgbkrk/atom-script/blob/master/README.md>
- [36] LEVENT-LEVI, Thasi. What is WebRTC and What is it Good For? Bloggeek.me [online]. c2020, 15. 12. 2019 [cit. 2020-05-07]. Dostupné z: <https://bloggeek.me/what-is-webrtc/>
- [37] SULLIVAN, Russell. CRDTs explained: Supercharge your serverless with CRDTs at the Edge. SERVERLESS, INC. Serverless [online]. c2020, 13. 11. 2017 [cit. 2020-05-07]. Dostupné z: <https://www.serverless.com/blog/crdt-explained-supercharge-serverless-at-edge/>

Seznam příloh

latex_test.tex
pokus_matlab.m
pokus_octave.m
pokus_scilab.sci

Testovací soubor Atom-Latex
Testovací soubor Atom-MATLAB
Testovací soubor Atom-Octave
Testovací soubor Atom-Scilab